Using Application-Specific Performance Models to Inform Dynamic Scheduling

Jeffrey Vetter*[1]

[1]Oak Ridge National Laboratory (ORNL) – Oak Ridge National Laboratory P.O. Box 2008 Oak Ridge, TN 37831, United States

Abstract

Concerns about energy-efficiency and reliability have forced our community to reexamine the full spectrum of architectures, software, and algorithms that constitute our ecosystem. While architectures have remained relatively stable for almost two decades, new architectural features, such as heterogeneous processing, nonvolatile memory, and optical interconnection networks, will demand that software systems and applications be redesigned to exploit these new capabilities. A key capability of this activity is accurate modeling of performance, power, and resiliency. We have developed the Aspen performance modeling language that allows fast exploration of the design space. Aspen is a domain specific language for structured analytical modeling of applications and architectures. Aspen is designed to enable rapid exploration of new algorithm and architectures. Because of the succinctness, expressiveness, and composability of Aspen, it can be used to model many properties of a system including performance, power, and resiliency. Aspen has been used to model traditional HPC applications, and recently extended to model scientific workflows for HPC systems and scientific instruments, like ORNL's Spallation Neutron Source. Models can be written manually or automatically generated from other structured representations, such as application source code or execution DAGs. These Aspen models can then be used for a variety of purposes including predicting performance of future applications, evaluating system architectures, informing runtime scheduling decisions, and identifying system anomalies. Aspen is joint work with Jeremy Meredith (ORNL).

*Speaker